4

# UNITED STATES PATENT AND TRADEMARK OFFICE

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/729,853 | 12/05/2003 | Relja Ivanovic | MSFT120968 | 8313 |

28319      7590      01/26/2007
BANNER & WITCOFF LTD.,·
ATTORNEYS FOR CLIENT NOS. 003797 & 013797
1001 G STREET , N.W.
SUITE 1100
WASHINGTON, DC 20001-4597

| EXAMINER |
|---|
| TERMANINI, SAMIR |

| ART UNIT | · PAPER NUMBER |
|---|---|
| 2178. | |

| SHORTENED STATUTORY PERIOD OF RESPONSE | MAIL DATE | DELIVERY MODE |
|---|---|---|
| 3 MONTHS | 01/26/2007 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

PTOL-90A (Rev. 10/06)

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on _05 December 2003_.

2a)☐ This action is **FINAL.**      2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) _1-41_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-41_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _12/5/03_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All  b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _3/26/04_.

4)☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5)☐ Notice of Informal Patent Application

6)☐ Other: _____.

# DETAILED ACTION

## BACKGROUND

1.      This action is responsive to the following communications: Application filed on 12/5/2003.

2.      Claims 1-41 are pending in this case. Claims 1, 12, 19, 25, 33, 38 are in independent form.

3.      The information disclosure statement (IDS) filed on 3/26/2004 has been acknowledged and considered by the examiner. The Initial copy of form PTO-1449 is included in this office action.

## CLAIM REJECTIONS - 35 U.S.C. §112

4.      The following is a quotation of the second para. of 35 U.S.C. §112:

```
The specification shall conclude with one or more claims particularly pointing
out and distinctly claiming the subject matter which the applicant regards as
his invention.
```

5.      Claim 9 rejected under 35 U.S.C. 112, 2ND paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Regarding claim 9, the phrase "such as" renders the claim indefinite because it is unclear whether the limitations following the phrase are part of the claimed invention. See MPEP § 2173.05(d).

## CLAIM REJECTIONS - 35 U.S.C. §102

6.      The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –
(a) the invention was known or used by others in this country, or patented or
described in a printed publication in this or a foreign country, before the
invention thereof by the applicant for a patent.

7.      **Claims 1-7, 9-20, 22-24, 33-34, and 36-41** are rejected under 35 U.S.C. §102(a)

as being anticipated by *ContainerListView and TreeListView: Writing VS.NET design-*

*surface compatible controls*, Jon *Rista* (1/14/2003)(hereinafter "*Rista*").

As to independent **claim 1**, *Rista* teaches a method for representing a property on a

display, comprising: a drawing handler being assigned to the property (e.g. "new

ContainerSubListViewItem("Test"), p. 3); the drawing handler being called to provide a

graphical representation of the property (e.g. progress bar, p. 3); and the drawing handler

converting a value for the property into a graphical representation which is provided on the

display (e.g. value "pb.Value = 25" is represented as the progress bar for test.ede in the Fig.

At the top of p. 1).

As to dependent **claim 2**, *Rista* further teaches that the graphical representation

includes a number of elements (the progress bar is broken up into elements, see Fig. on

page 1), and the number of elements is proportional to a numerical value of the property

(e.g. value "pb.Value = 25").

As to dependent **claim 3**, *Rista* further teaches that the drawing handler component

is registered independently of the other system components ("an instance of an object," p. 7;

see also without hooking into the Windows Common Controls (as the standard ListView

does) or using Windows API calls, p. 3 and top of p. 4).

As to dependent **claim 4**, *Rista* further teaches the drawing handler component is

called by an implementation component ("ContainerListView" p. 3).

As to dependent **claim 5**, *Rista* further teaches that a list view component which is responsible for the presentation on the display communicates with the implementation component ("The ContainerListView provides the ability to embed text, an image, or a control into each subitem of a ListView item." p. 3).

As to dependent **claim 6**, *Rista* further teaches that the list view component has different modes, including at least one mode for which the drawing handler component may be required, and at least one mode for which the drawing handler component will not be required (The Extended ListView Test on p. 1, while utilizing the TreeListView does not require the drawing handler component).

As to dependent **claim 7**, *Rista* further teaches that the list view component has a details mode in which the drawing handler component may be required, but in which the user is able to cancel the column that would otherwise require the drawing handler component (by clicking on the The Extended ListView Test, see Fig. on p. 1).

As to dependent **claim 9**, *Rista* further illustrates on p. 2, within the size column, that each indicator is related to the physical property of the media being listed within each of the individual rows.

As to dependent **claims 10 and 11**, *Rista* further teaches, inter alia, within the figure at the top of page 2 that a plurality of drawing handlers are assigned to a plurality of different properties. Additionally, the "video file" shown in the last row of the extended ListView test has a property value much higher than the "document" shown in the third row of the extended ListView test where separate property values and drawing handlers, are used.

As to independent **claim 12**, *Rista* teach a method for representing a property on a display (see property "pb.Value=25," bottom part of p. 3) comprising: receiving a call to provide a graphical representation of the property value (see "Custom control rendering" p.4) and in response to receiving the call, providing a graphical representation that corresponds to the property value (See p. 2).

As to dependent **claims 13 and 14**, *Rista* et al. further teach, that the graphical representation includes a number of elements, in this instance a plurality of bars (p. 1), where the number of bars correspond proportionality to the numeric value of a rating (i.e capacity rating - in size).

As to dependent **claim 15**, *Rista* further teach that the property represents physical characteristic of the item (see physical file "size", figure at top of page 2). He

As to dependent **claim 16**, *Rista* teach that the graphical representation is produced by a drawing handler component (e.g. "new ContainerSubListViewItem("Test"), p. 3).

As to dependent **claim 17**, *Rista* teach that a implementation component makes the call to the drawing handler component ("ContainerListView" p. 3).

As to dependent **claim 18**, *Rista* teach wherein a list view component communicates with the implementation component for providing information on the display, component ("The ContainerListView provides the ability to embed text, an image, or a control into each subitem of a ListView item." p. 3).

As to independent **claim 19**, *Rista* teach inter-component communication ("The ContainerListView provides the ability to embed text, an image, or a control into each subitem of a ListView item." p. 3), for providing a representation of a property on a display (p. 2), said media comprising: a set of computer-usable instructions that cause a request to

provide a graphical representation of a property to be communicated to one or more other computer-program segments capable of executing said request (see code on p. 3 and p. 7-8).

As to dependent **claim 20,** *Rista* teach that the graphical representation includes a number of elements, the number of elements (the progress bar is broken up into elements, see Fig.1 on page 1) being proportional to the numerical value of the property (e.g. value "pb.Value = 25"; see also "size," p. 2).

As to dependent **claim 22,** *Rista* further teach that the property represents a physical characteristic of an item (within the size column, each graphical indicator is related to the physical property of the media being listed within each of the individual rows, p. 2.)

As to dependent **claim 23,** *Rista* further teach that the process segments are capable of executing the request for producing a graphical representation comprise a drawing handler component (e.g. "ContainerListView" p. 3).

As to dependent **claim 24,** *Rista* teach the drawing handler component to be independently registered ("an instance of an object," p. 7; see also "without hooking into the Windows Common Controls (as the standard ListView does) or using Windows API calls," p. 3 and top of p. 4).

As to independent **claim 33,** *Rista* teach representing a property (see property "pb.Value=25," bottom part of p. 3) on a display, comprising: a set of computer-usable instructions that are independently registered (without hooking into the Windows Common Controls (as the standard ListView does) or using Windows API calls, p. 3 and top of p. 4) and which operate to produce a graphical representation of the property on the display (See display in p. 2).

As to dependent **claim 34**, *Rista* teach that the graphical representation includes a number of elements (the progress bar is broken up into elements, see Fig. on p. 2), the number of elements corresponding to the numerical value of the property (e.g. value "pb.Value = 25"; the larger this value, the greater the number of corresponding elements).

As to dependent **claim 36**, *Rista* teach the indicator is related to the physical property of the media being listed within each of the individual rows (e.g. physical "size", see p. 2).

As to dependent **claim 37**, *Rista* teach The media of claim 33, wherein the computer-usable instructions are provided as part of a drawing handler component (e.g. "ContainerListView" p. 3).

As to independent **claim 38**, *Rista* teach providing a representation of a property on a display, comprising: an implementation component providing to a drawing handler component a numerical value corresponding to the property (e.g. value "pb.Value = 25"); and the drawing handler component converting the numerical value into a graphical representation for the property (e.g. "new ContainerSubListViewItem("Test"), p. 3); see also progress bar, p. 3).

As to dependent **claim 39**, *Rista* teach 39. The method of claim 38, wherein the drawing handler component is registered independently of the implementation component ("an instance of an object," p. 7; see also without hooking into the Windows Common Controls (as the standard ListView does) or using Windows API calls, p. 3 and top of p. 4).

As to dependent **claim 40**, *Rista* teach 40. The method of claim 38, wherein a list view component communicates with the implementation component and is responsible for

the organization of items on the display ("The ContainerListView provides the ability to

embed text, an image, or a control into each subitem of a ListView item." p. 3).

As to dependent claim 41, *Rista* teach 41. The method of claim 40, wherein the list

view component has a plurality of modes, including at least one mode for which the drawing

handler component may be required and at least one mode for which the drawing handler

component is not required (The Extended ListView Test on p. 1, while utilizing the

TreeListView does not require the drawing handler component).


8.      The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102

that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
> (b) the invention was patented or described in a printed publication in this or
> a foreign country or in public use or on sale in this country, more than one
> year prior to the date of application for patent in the United States.

9.      Claims 25-32 are rejected under 35 U.S.C. 102(b) as being anticipated by

*Neat Stuff to do in List Controls Using Custom Draw*, Michael *Heller et al.*

(11/30/1999)(hereinafter *"Heller et al."*).

As to dependent claim 25, *Heller et al.* teaches a method of representing a property

on a display (e.g. individual image, bottom of p. 8), comprising: an implementation

component issues a call for providing a graphical representation of a property on the

display ("The NM_CUSTOMDRAW message passes" p. 3); and in response to the call, a

drawing handler component produces the graphical representation for the property on the

display ("NMLVCUSTOMDRAW struct" p. 3).

As to dependent claim 26, *Heller et al.* teaches that the graphical representation

includes a number of elements, the number of elements being proportional to the numeric

value of the property (index number are proportional to the number of items being drawn, middle of p. 3).

As to dependent **claim 27**, *Heller et al.* teaches that the drawing handler component is independently registered ("NM_CUSTOMDRAW messages for each item" p. 3; see also "as if there were no custom draw handler," p.2).

As to dependent **claim 28**, *Heller et al.* teaches 28. The method of claim 25, wherein the drawing handler component may be modified without requiring modifications to other system components ("You can choose to ignore the notifications altogether (in which case you'll see the standard list control), process some part of the drawing yourself (for simple effects), or even draw the control yourself (just as when owner-drawing a control)." p. 1).

As to dependent **claim 29**, *Heller et al.* teaches that a list view component communicates with the implementation component and organizes the presentation of information on the display ("handle a WM_NOTIFY message sent from your list control," top of p. 2).

As to dependent **claim 30**, *Heller et al.* teaches that the list view component has different modes, including at least one mode for which the drawing handler component may be required ("I don't want to do anything now; Windows should paint the control or item itself as if there were no custom draw handler." p. 2), and at least one mode for which the drawing handler component is not required ("I want to receive additional NM_CUSTOMDRAW messages during the draw stages of each subitem in the row currently being drawn." p. 2).

As to dependent claim 31, *Heller et al.* teaches that the list view component has a details mode, in which the user may select or cancel a column which requires the use of the drawing handler component (e.g. "report view mode with multiple columns" p. 1).

As to dependent claim 32, *Heller et al.* teaches that a plurality of drawing handler components ("requesting notifications on a per-item basis" top of p. 3) are utilized for creating graphical representations for a plurality of properties. (e.g. "Flags indicating the state of the item being drawn (selected, grayed, etc.)," p. 3).

CLAIM REJECTIONS - 35 USC §103

10.     The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

11.     Claims 8, 21 and 35 are rejected under 35 U.S.C. 103(a) as being unpatentable over *Rista* in view of *Heller et al.*

As to dependent claim 8, *Rista* teaches all of the limitations previously addressed in the treatment of claim 1. However, *Rista* fails to teach that the property represents a rating value of an item. *Heller et al.* is cited for teaching that the property represents a rating value of an item ("rating," para. [0048]).

It would have been obvious to one of ordinary skill in the art at the time the invention was made, to have the property value representative of an item rating, as taught in *Heller et al.*, be used in the extended ListView of *Rista*, because *Heller et al.* teach the

automatic updating of a list view of the items when it is determined that at least one of them has been altered ("the playlist is redrawn space 536," para. [0057] via a e.g. "message receiver," para. [0049]; see also para. [0013]) and further suggest that the same type of automatic updating of ListView's to avoid the disadvantage of the user having to manually update items in a list.

As to substantially identical dependent **claims 21 and 35,** *Rista* teaches all of the claims' limitations as previously addressed, above. However, *Rista* fails to teach that the property represents a rating value of an item. *Heller et al.* is cited for teaching that the property represents a rating value of an item ("rating," para. [0048]; see also Fig. 11A and 11B).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to have the property represent a rating value of an item as taught in *Heller et al.* in the extended ListView of *Rista* because *Heller et al.* suggest a list view ("a list 1104" para. [0086]) where the visual indicator is a "'My Rating'" field for each of the media items in the list 1104" (para. [0086]), where the property of the graphical representation represents a rating value of an item, specifically: a rating. *Heller et al.* further suggest utilizing the graphical property display ("filter conditions are based on a user rating of the media items (e.g., 1, 2, 3, 4 or 5 star rating" para. [0084]) a desirable advantage in that it better helps the user manage media content (facilitates creation of...playlist[s]," para. [0084]).

## CONCLUSION

12.     Although not relied upon, the following prior art is made of record because it

considered pertinent to applicant's disclosure:

[1]     Aalbersberg (US 5946678 A) for teaching a user interface where each
        document header or title or representation is accompanied by an indicator
        which employs the same distinctive representation to directly indicate to the
        user the relative contributions of the individual query words to each listed
        document.

[2]     Aalbersberg (US 6094648 A) for teaching a user interface with a first
        indicator in a words window, and in the results window distinctive
        representations.

[3]     Burner et al. (US 6282548 B1) for teaching a button bar in conjunction with
        the web page display.

[4]     Blower et al. (US 6323852 B1) for teaching a panel showing a graph (e. g., a
        bar graph) of the contents of a set shown in the first panel, the size of the bar
        graph may represent the number of members of the set containing the
        Feature or the biological or physical property of the set.

[5]     Helsper et al. (US 20020049687 A1) for teaching a graphical status of
        underlying indicators.

[6]     Couchot et al. (US 20030078756 A1) for teaching a metric catalog comprising
        a number of metrics, where each metric is associated with a threshold value.

[7]     Burke (US 20040070594 A1) for teaching classification codes to identify a
        class to which an associated programme element has been allocated.

[8]     Scozzafava (US 20040217228 A1) for teaching the display of user defined
        information.

[9]     Gatto (US 7149716 B2) for teaching different ranges of scores being assigned
        corresponding graphical symbols or other indicators of relative accuracy.
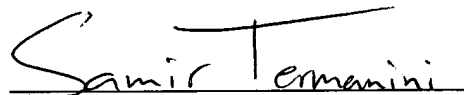
13.     Any inquiry concerning this communication or earlier communications from

the Examiner should be directed to Samir Termanini whose telephone number is (571) 270-

1047. The Examiner can normally be reached from 9 A.M. to 4 P.M., Monday through Friday (excluding alternating Fridays).

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Stephen S. Hong can be reached on (571) 272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

STEPHEN HONG
SUPERVISORY PATENT EXAMINER

Samir Termanini
Patent Examiner
Art Unit 2178